# JAVA MINI PROJECT

# FLAPPY BIRD

**Sarvagya Singh (60009200030)**
**Aditya Potdar (60009200027)**
**Shyam Bhuva (60009200018)**

## Code:

## FlappyBird.java

```
package flappyBird;

import java.awt.Color;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.Rectangle;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;
import java.util.ArrayList;
import java.util.Random;

import javax.swing.JFrame;
import javax.swing.Timer;

public class FlappyBird implements ActionListener,
MouseListener, KeyListener
{

  public static FlappyBird flappyBird;

  public final int WIDTH = 800, HEIGHT = 800;

  public Renderer renderer;

  public Rectangle bird;

  public ArrayList<Rectangle> columns;
```

```
  public int ticks, yMotion, score;

  public boolean gameOver, started;

  public Random rand;

  public FlappyBird()
  {
    JFrame jframe = new JFrame();
    Timer timer = new Timer(20, this);

    renderer = new Renderer();
    rand = new Random();

    jframe.add(renderer);
    jframe.setTitle("Flappy Bird");
    jframe.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    jframe.setSize(WIDTH, HEIGHT);
    jframe.addMouseListener(this);
    jframe.addKeyListener(this);
    jframe.setResizable(false);
    jframe.setVisible(true);

    bird = new Rectangle(WIDTH / 2 - 10, HEIGHT / 2 - 10, 20,
20);
    columns = new ArrayList<Rectangle>();

    addColumn(true);
    addColumn(true);
    addColumn(true);
    addColumn(true);

    timer.start();
  }

  public void addColumn(boolean start)
  {
    int space = 300;
    int width = 100;
    int height = 50 + rand.nextInt(300);

    if (start)
    {
      columns.add(new Rectangle(WIDTH + width + columns.size() *
300, HEIGHT - height - 120, width, height));
      columns.add(new Rectangle(WIDTH + width + (columns.size()
- 1) * 300, 0, width, HEIGHT - height - space));
```

```
    }
    else
    {
      columns.add(new Rectangle(columns.get(columns.size() -
1).x + 600, HEIGHT - height - 120, width, height));
      columns.add(new Rectangle(columns.get(columns.size() -
1).x, 0, width, HEIGHT - height - space));
    }
  }

  public void paintColumn(Graphics g, Rectangle column)
  {
    g.setColor(Color.green.darker());
    g.fillRect(column.x, column.y, column.width, column.height);
  }

  public void jump()
  {
    if (gameOver)
    {
      bird = new Rectangle(WIDTH / 2 - 10, HEIGHT / 2 - 10, 20,
20);
      columns.clear();
      yMotion = 0;
      score = 0;

      addColumn(true);
      addColumn(true);
      addColumn(true);
      addColumn(true);

      gameOver = false;
    }

    if (!started)
    {
      started = true;
    }
    else if (!gameOver)
    {
      if (yMotion > 0)
      {
        yMotion = 0;
      }

      yMotion -= 10;
    }
  }
```

```
  @Override
  public void actionPerformed(ActionEvent e)
  {
    int speed = 10;

    ticks++;

    if (started)
    {
      for (int i = 0; i < columns.size(); i++)
      {
        Rectangle column = columns.get(i);

        column.x -= speed;
      }

      if (ticks % 2 == 0 && yMotion < 15)
      {
        yMotion += 2;
      }

      for (int i = 0; i < columns.size(); i++)
      {
        Rectangle column = columns.get(i);

        if (column.x + column.width < 0)
        {
          columns.remove(column);

          if (column.y == 0)
          {
            addColumn(false);
          }
        }
      }

      bird.y += yMotion;

      for (Rectangle column : columns)
      {
        if (column.y == 0 && bird.x + bird.width / 2 > column.x
+ column.width / 2 - 10 && bird.x + bird.width / 2 < column.x +
column.width / 2 + 10)
        {
          score++;
        }
```

```
        if (column.intersects(bird))
        {
          gameOver = true;

          if (bird.x <= column.x)
          {
            bird.x = column.x - bird.width;

          }
          else
          {
            if (column.y != 0)
            {
              bird.y = column.y - bird.height;
            }
            else if (bird.y < column.height)
            {
              bird.y = column.height;
            }
          }
        }
      }

      if (bird.y > HEIGHT - 120 || bird.y < 0)
      {
        gameOver = true;
      }

      if (bird.y + yMotion >= HEIGHT - 120)
      {
        bird.y = HEIGHT - 120 - bird.height;
        gameOver = true;
      }
    }

    renderer.repaint();
  }

  public void repaint(Graphics g)
  {
    g.setColor(Color.cyan);
    g.fillRect(0, 0, WIDTH, HEIGHT);

    g.setColor(Color.orange);
    g.fillRect(0, HEIGHT - 120, WIDTH, 120);

    g.setColor(Color.green);
    g.fillRect(0, HEIGHT - 120, WIDTH, 20);
```

```
    g.setColor(Color.red);
    g.fillRect(bird.x, bird.y, bird.width, bird.height);

    for (Rectangle column : columns)
    {
      paintColumn(g, column);
    }

    g.setColor(Color.white);
    g.setFont(new Font("Arial", 1, 100));

    if (!started)
    {
      g.drawString("Click to start!", 75, HEIGHT / 2 - 50);
    }

    if (gameOver)
    {
      g.drawString("Game Over!", 100, HEIGHT / 2 - 50);
    }

    if (!gameOver && started)
    {
      g.drawString(String.valueOf(score), WIDTH / 2 - 25, 100);
    }
  }

  public static void main(String[] args)
  {
    flappyBird = new FlappyBird();
  }

  @Override
  public void mouseClicked(MouseEvent e)
  {
    jump();
  }

  @Override
  public void keyReleased(KeyEvent e)
  {
    if (e.getKeyCode() == KeyEvent.VK_SPACE)
    {
      jump();
    }
  }
```

```
  @Override
  public void mousePressed(MouseEvent e)
  {
  }

  @Override
  public void mouseReleased(MouseEvent e)
  {
  }

  @Override
  public void mouseEntered(MouseEvent e)
  {
  }

  @Override
  public void mouseExited(MouseEvent e)
  {
  }

  @Override
  public void keyTyped(KeyEvent e)
  {

  }

  @Override
  public void keyPressed(KeyEvent e)
  {

  }

}
```

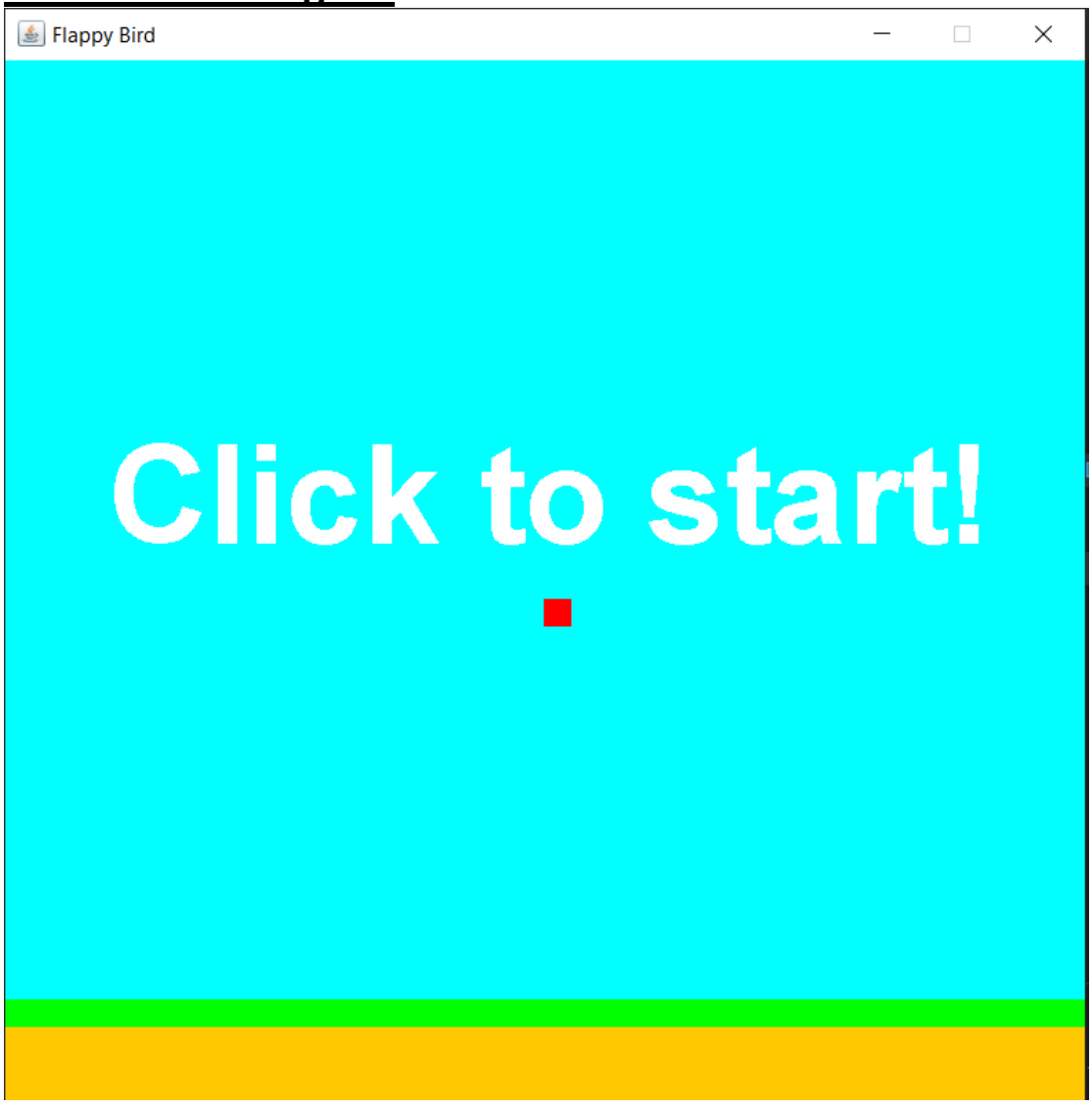
**Renderer.java**

```
package flappyBird;

import java.awt.Graphics;

import javax.swing.JPanel;

public class Renderer extends JPanel
{

  private static final long serialVersionUID = 1L;

  @Override
```

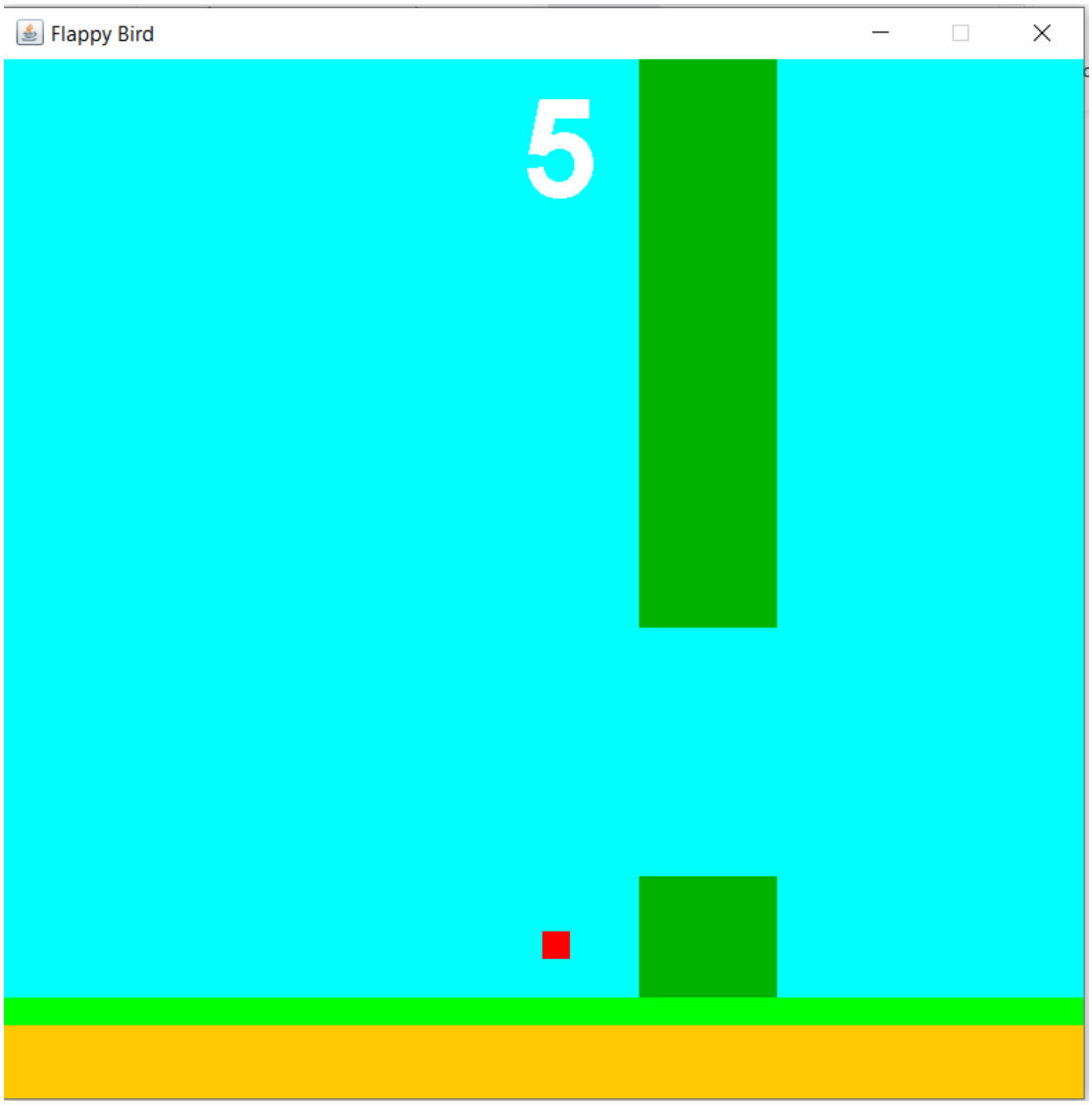
```
  protected void paintComponent(Graphics g)
  {
    super.paintComponent(g);

    FlappyBird.flappyBird.repaint(g);
  }

}
```

## Output:

## Start of the game




**Press Space bar to start playing**




**Press Space bar to Jump**

**After hitting any column it shows (“GAME OVER!”)**